LEGIBILITY NOTICE

A major purpose of the Technical Information Center is to provide the broadest dissemination possible of information contained in DOE's Research and Development Reports to business, industry, the academic community, and federal, state and local governments.

Although a small portion of this report is not reproducible, it is being made available to expedite the availability of information on the research discussed herein.

1

LA-UR -87-3129

CONF-880538--1

LA-UR--87-3129

DE88 000500

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

TITLE THE -MDOC MACRO PACKAGE: A SOFTWARE TOCL TO SUPPORT COMPUTER DOCUMENTATION STANDARDS

AUTHOR(S) Claudia E. Sanders, C-2

SUBMITTED TO 35th Annual International Technical Communication Conference
Philadelphia, PA
May 10-13, 1988

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Los Alamos
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

The -mdoc Macro Package: A Software Tool to Support Computer Documentation Standards

Claudia E. Sanders*
Computer Documentation Group (C-2)
Computing and Communication Division
Los Alamos National Laboratory
Los Alamos, NM 87545

ABSTRACT

At Los Alamos National Laboratory a small staff of writers and word processors in the Computer Documentation Group is responsible for producing computer documentation for the over 8000 users of the Laboratory's computer network. The -mdoc macro package was developed as a software tool to support that effort. The -mdoc macro package is used with the NROFF/TROFF document preparation system on the UNIX operating system.

The -mdoc macro package incorporates the standards for computer documentation at Los Alamos that were established by the writers. Use of the -mdoc macro package has freed the staff of programming format details, allowing writers to concentrate on content of documents and word processors to produce documents in a timely manner. It is an easy-to-use software tool that adapts to changing skills, needs, and technology.

1. INTRODUCTION

Los Alamos National Laboratory is the home of a complex scientific computer network that serves over 8000 users. While many of the users are located at Los Alamos, there are over 3000 users at remote sites. Our mission in the Computer Documentation Group is to provide user documentation to all users in a timely, efficient manner. Seven technical writers and six word processors strive to keep over 300 locally produced documents up to date, issue two publications every month, develop over 20 new documents every year, prepare several annual reports, and edit and review a multitude of other documents. There are two programmers to support this effort.

1.1. How We Got Where We Are

In 1976 the UNIX operating system was evaluated at Los Alamos and was found to be a desirable system for text editing and document preparation. Since that time the NROFF/TROFF document preparation system has been used by the Computer Documentation Group to produce a variety of documents and reports.

By 1983 the computer network had become so complex that it was necessary to evaluate how to provide useful computer documentation to the users when they needed it. As a result, the Computer Documentation Group published a documentation plan that determined the types of documentation needed to fulfill user needs. The

*This work was performed under the auspices of the U.S. Department of Energy.

types of documentation were based on three user attitudes identified as

- no time to learn,
- want to learn, and
- know what I want.

To satisfy these needs, four principal categories of documentation were defined as:

- catalog for everyone,
- quick steps for users with no time to learn,
- learning for users who want to learn, and
- reference for users who know what they want.

As part of the documentation plan, the Computer Documentation Group prepared prototypes of catalog, reference, and quick steps documentation. Each prototype was prepared using both NROFF and TROFF.

NROFF formats in a typewriter style that can be printed on a line printer. TROFF formats in a typesetter style that can be printed on a variety of printers or viewed at a terminal. NROFF and TROFF accept the same formatting commands but NROFF ignores the ones that specify font and point size changes. Therefore, the same source file can be used to produce the printed and the online versions of a document. NROFF output is stored online for retrieval by users to view at their terminals or print on their local line printers. TROFF output is used for the hardcopy documents that are distributed through the Computer Information Center at Los Alamos.

1.2. What was the Problem?

To generate the prototypes, macro packages were written that contained commands specific to Los Alamos documents as well as to three of the four principal categories of documents. A macro package contains many macro definitions. A macro definition gives a name to the macro and a sequence of instructions (in this case, formatting instructions) to be followed when that macro name is used in the source file.

Since technical writers and word processors share the task of producing documents, more than one person often works on the same document. Therefore, each writer and word processor kept a personal copy of the macro package being used for each document that he or she was working on. Changes were often made to the macro packages. Thus more varieties of macro packages were introduced—some through necessity, some through ignorance of what was already available. The propagation of so many macro packages caused repetitive work, inconsistent documents, and loss of productivity.

1.3. What was Proposed?

To solve this problem the management of the Computer Documentation Group proposed a new project. The project was to produce a single macro package that would be used for all documents and stored in a central location on the computer. Furthermore, it was thought that such a macro package would be a powerful software tool that would relieve the non-programmer staff of the frustration of learning in depth the programming language being used. I was assigned the project of developing the macro package that resulted in the -mdoc macro package.

2. SPECIFICATIONS OF THE NEW MACRO PROJECT

The goal of the project was to maximize productivity of the limited number on the staff by minimizing time spent formatting documents. Use of the macro package would result in the production of high-quality, consistent documentation in a timely and efficient manner.

The project was structured in the following phases.

- Study macro packages being used.
- Determine formatting needs of writers and word processors.
- Write, test, and debug the basic package.
- Maintain and add enhancements.

2.1. Phase 1: Study Current Macro Packages

When I began the project I was a novice NROFF/TROFF user. I was able to gain knowledge of the formatter by studying the macro packages and debugging formatting problems for the writers and word processors.

While studying the packages being used I discovered there were still three basic packages, but with many variations. The variations were between basic types and between copies used by writers and word processors. Often the differences between packages were trivial, such as spacing, print style, and print size, but these trivial differences had destroyed the consistent look that represented dependability and credibility in the locally produced documentation.

I realized the importance of providing a declarative language interface to the writers and word processors. Procedural language describes *how* to format a document and declarative language describes *parts* of a document. As the programmer I would use procedural anguage to write the macros so that the writers and word processors could use declarative language. For example, I wrote a macro that contained 50 lines of procedural commands that tell NROFF/TROFF *how* to start a new chapter. The writer would use one declarative command, .CH title, to tell NROFF/TROFF the *part* of the document that would be formatted next.

2.2. Phase 2: Determine Formatting Needs

I talked with the writers individually to discover their expectations of what the macro package should provide. It became obvious that a basic problem would be centered around writers preferring to retain individuality in their style and the need for documents to be identified with the group rather than one individual writer. In preparing the macros for the -mdoc package, I needed to provide mechanisms that would allow the writer some individual style, yet would retain the consistency of the documentation produced in the group.

While the writers all agreed that standards were needed, there was a reluctance by many individual writers to give up creative freedom even within format specifications. My area was programming and not writing, so it would be up to the team of writers to establish a set of standards, a style guide, and review procedures to ensure that the documentation produced by the group was a high-quality product.

With knowledge of these needs, I decided that the -mdoc macro package would

- provide declarative language for the writers and word processors,
- be used for all types of documents including the online versions,
- be " upward compatible," that is, existing source files would not become obsolete by updated versions of -mdoc, and
- produce documents easily identifiable as documentation produced by our group based on a set of standards used by all the writers

2.3. Phase 3: Code and Test Macro Package

With all the preliminaries completed, it was time to begin coding the macro package. First I collected a set of macros representing the different macros from all the packages then in use. This was the package that would be the basis for -mdoc. Since the initial -mdoc package contained all the macros that the staff was familiar with in their previous documentation work, it would be easy for them to use. It also meant existing files could be updated with minimum effort. Some of the basic macros generate chapter titles, section titles, headers and footers for even and odd pages, boxes for notes, delimiters for examples, bulleted items, glossaries, and a table of contents.

To make the macro package work for all types of documents, I added a macro that would be used to flag the type of document being created, for example, reference, quick steps, or catalog. With that done I could modify the basic macros to be used with all documents. Next I included formatting instructions that applied only when the document was being formatted for online use.

The NROFF/TROFF document preparation system has some standard macro packages distributed with it, usually -me or -ms. Members of the Computer Documentation Group had always used the standard -me macro package, so I added a link to that information in the -mdoc package so the -me macros could be used while using -mdoc.

As each macro was developed or changed, it was tested. I prepared a document about -mdoc that could be used as a reference manual by the computer documentation staff (Ref. 1). Writing the reference manual increased my awareness of the problems the writers and word processors encountered when formatting their documents. When I believed the -mdoc package was almost ready for use, I asked a word processor to use it with a new document that was being created. This gave me an opportunity to correct problems before production use.

To avoid the previous problem of multiple copies being made by everyone, I asked the UNIX system manager to place the -mdoc macro package in the same directory as the standard macro packages. To follow the UNIX convention of the macro packages using the -m option, I named the new macro package -mdoc, -m for the option, doc for documentation. With the manual ready for use and the package in place, the writers and word processors began incorporating -mdoc in new documents as they were created and existing documents as they were updated. To my surprise, but perhaps because I was thorough in my testing, production use of -mdoc was almost free of bugs. The greatest "bug" was the request for more features. Soon I discovered that other users were using -mdoc to format user-supported software documentation and reports. I felt I had a good basic product at this point and was now ready to look at enhancements.

2.4. Phase 4: Modifications and Enhancements

As the writers became more confident using -mdoc, they also became more interested in establishing more detailed standards. Freed of worries about formatting details, they were able to be more concerned with content and quality of the group's products.

Often a writer would ask if I would change -mdoc to support a particular detail for a specific document. I adopted the stand that I would only make a change when requested by a majority of the writers. I also would not make changes without warning. I established the procedure of making a new version available for use for two weeks before the current version was replaced. That gives the writers time to test new features. Sometimes a bug is discovered and can be fixed; if not, the production date for a new version is delayed.

Soon the writers began a process of establishing precise standards for Los Alamos computer documentation. They held many meetings discussing standards to minute details. I attended some of their meetings to answer questions about whether -mdoc could be programmed to provide some particular feature. The project leader acted as the liaison between the writers and me so that I knew what changes to make to -mdoc to reflect the new standards.

The original -mdoc package contained 20 macros. The latest version contains 48 macros. Some of the macros added to the basic package include automatic indexing, merging of text and graphics, and boilerplate macros for standard forms and information used in all documents.

It is interesting to note that when -mdoc was first used in a production environment in 1984, the group did its work on terminals connected to a single VAX minicomputer. Output was obtained from an inkjet printer for NROFF output and a laser printer for TROFF output. As of fall 1987, the group still works on terminals, but has several VAX minicomputers to choose from. We also work on scientific workstations and we have several laser printers available. The -mdoc macro package has adapted well to changes in technology, needs, and skill.

3. EXAMPLES OF -mdoc FEATURES

The features of -mdoc can best be demonstrated with examples of how a few of the macros work. The examples also show the importance of providing a declarative language interface.

First let me offer a few words of explanation about the rules for NROFF/TROFF formatting commands. Formatting commands consist of a period followed by one or two characters. Sometimes called "dot commands," the period must always appear in the first column of an input line. Additional information, called "arguments," may be included on the same line with the dot commands. The macros and arguments for -mdoc are described in a reference manual (Ref. 1) as well as in a short reference guide, which are available to the users.

3.1. Examples of Macros with Optional Arguments

Earlier I referred to a .CH macro. To begin a new chapter it is only necessary to include a command line containing the macro name and the title of the chapter. This is what a command might look like.

.CH "How to Use the -mdoc Macros"

Figure 1 shows what would be produced by the NROFF/TROFF formatter for a reference or catalog document. Chapter headings always begin on a new page. The title of the chapter is automatically numbered and printed in all uppercase characters. The point size is larger and the print style is boldface. A horizontal line is added and helps to demarcate a chapter heading. In addition, a table of contents entry is saved to be used when the table of contents is generated for the document. For quick steps documents, the chapter would be unnumbered but in larger type. Optional arguments may be used to specify a page number, chapter number, and to retain the case of the title as given in the argument.

On the next line following the .CH command, the writer would begin entering the text of the first paragraph.

This example shows the benefits to the user of providing a declarative language interface. By *declaring* that a document part (in this case, chapter) is to be generated at that point in the document, the user only needs to give the chapter title to get all the formatting done for the beginning of a new chapter.

Figure 2 shows a standard cover that is generated by using the .CV macro. The arguments included with the CV macro are title, Computing Information Catalog numbers, and an optional date. The current date is used as the default.

Reduced figure here

Figure 1. Example of a Chapter Heading generated using the .CH macro.

Reduced figure here

Figure 2. Example of a Cover generated using the .CV macro.

3.2. Examples of Boilerplate Macros

Boilerplate macros are those that generate standard pieces of information. There are no arguments with these macros. For example, each of our computer documents contains a "Who to Contact" table that acts as a guide to the user to find help when needed.

There are many entries with associated phone numbers. This table has more than 200 lines of procedural code in the macro but is available by inserting only .CO in the source file at the place that it is to appear. Figure 3 shows what is produced when .CO is used.

Other boilerplate macros are used for forms included in documents. Every document has a form requesting reader comments. Figure 4 shows the form that is generated by inserting the macro .CM in the proper place in the file. The form will be printed on an odd numbered page, and the even numbered page (which appears on the reverse side when printed double sided) will contain an address plus instructions to fold the page and use it as an envelope.

Reduced
figure
here

Figure 3. Example of a "Who to Contact" table generated using the .CO macro.

Reduced
figure
here

Figure 4. Example of a form generated using the .CM macro.

More examples of the macros are contained in the reference manual *-MDOC, Documentation Macros for NROFF/TROFF* (Ref. 1).

4. CONCLUSIONS

While I have described -mdoc as a tool specifically used with the NROFF/TROFF document preparation system the concepts can be applied to other systems. These concepts are

- make a software tool easy to use and
- make the software tool adaptable to changing technology, standards, and needs.

A software tool for computer documentation that is easy to use uses declarative language and keeps learning time to a minimum. An adaptable software tool is built on a well-designed base that may be improved as skill, technology, and knowledge change.

The addition of the -mdoc macro package to the tools used by the Computer Documentation Group has added to the efficiency and productivity of the staff. A high-quality product is being produced, and the writers can use more time for information gathering and writing and less time worrying about formatting problems. A common language is used between the writers, the word processors, and the programmer so less time is needed to discuss the formatting needs of a particular document.

The -mdoc macro package will continue to be maintained and enhanced as needed, but as it stands today, it is a powerful tool used to produce high-quality computer documentation at Los Alamos National Laboratory.

Bibliography

1. Claudia E. Sanders, *-MDOC, Documentation Macros for NROFF/TROFF*, Los Alamos National Laboratory internal document.
2. Computer Documentation Group, *Computing Division Plan for User Documentation*, Los Alamos National Laboratory report LA-9807-MS, July 1983.
3. Joseph F. Ossanna, *NROFF/TROFF User's Manual*, Bell Laboratories, published in *UNIX User's Manual, Supplementary Documents*, University of California, Berkeley, March 1984.
4. Kevin P. Roddy, *UNIX NROFF/TROFF: A User's Guide*, CBS College Publishing, 1987.
5. Eric P. Allman, *-ME Reference Manual*, published in *UNIX User's Manual, Supplementary Documents*, University of California, Berkeley, March 1984.

1. WHAT IS THIS DOCUMENT?

This document is a reference for the macro package for C-2 documentation. For some time many of these macros have existed in slightly different packages in several places. Now all of these packages have been incorporated into one. Subsequent improvements to these macros will be done using this as the basic package.

To use this package the user must use -mdoc on the NROFF/TROFF command line. An example of a command line is the following.

```
        qtroff -mdoc sourcefile
```

Mdoc calls the me macro package so -me should not be included on the command line. Mdoc also calls the -mix macro package that allows automatic indexing.

To send a job to receive output from PAGES (Printing and Graphics Express Station) use the following command line.

```
        ptroff -mdoc sourcefile
```

1.1. WHAT MAKES THIS WORK FOR ALL TYPES?

A macro called TY has been added to the macros. The second macro call of a file should be to this macro with an appropriate argument. The first call should be to the HE macro which is described later in the document. Currently, there are only three possible arguments. They are "Quick Steps", "Reference", and "Catalog".

```
.TY Reference
```

This macro serves as a signal to the rest of the macro package so that appropriate point sizes and other differences can be set according to the type of document that is being produced.

Figure 1

September 1987

Computing and Communications Division

-MDOC

Documentation Macros for NROFF/TROFF

Los Alamos

Los Alamos National Laboratory
Los Alamos,New Mexico 87545

Figure 2

WHO TO CONTACT

CONTACT	PHONE	FTS
Consulting		
ICN Consulting Office (C-10) - 6 a.m. to 6 p.m. Mountain Time ARPANET/MILNET mail address consult@lanl.gov	(505)667-5745	843-5745
CCF Dispatcher/CFS/PAGES (C-1)	(505)667-4584	843-4584
CAD/CAM Consulting (C-6)	(505)667-7356	843-7356
PCHELP (C-10)	(505)667-5884	843-5884
INFORM Consulting Office	(505)667-9444	843-9444
Telenet Customer Service	(800)336-0437	
Defense Data Network Consulting (SRI)	(800)235-3155	
Data Communications Service (C-4, Installation/Consulting)	(505)667-2686	843-2686
Communications Problems		
ICN Communications Problems (C-4 Trouble Desk)	(505)667-7423	843-7423
AT&T WATS Trouble Number	(800)222-3000	
Terminal Repair and Purchasing		
Terminal and Minicomputer Repair (MEC-9) - VT100 etc.	(505)667-5940	843-5940
IBM-PC Repair (MEC-9)	(505)667-9372	843-9372
Terminal Coordinator (C-DO selection and purchase assistance)	(505)667-6171	843-6171
Education and Documentation		
Computing Information Center - CIC (C-2 publication distribution) ARPANET/MILNET mail address cic@lanl.gov	(505)667-6992	843-6992
Computing Learning Center - CLC (C-2)	(505)667-8722	843-8722
LABNET Scheduling of Courses	(505)667-3193	843-3193
Status Messages		
CCF Status Recording	(505)667-5588	843-5588
FOCUS - Machine Status	(505)667-2852	843-2852
Security		
Computer and Information Security (OS-4)	(505)667-4844	843-4844

Revision History

Revised September 1986
Revised February 1985
Originally published, October 1984

Written by Claudia Sanders, Computer Documentation Group (C-2)
Originally published,

Figure 3

READER COMMENTS FORM

Date__________ Name______________________

Group__________ Mail Stop__________ Phone__________

Organization* ______________________

Street* ______________________

City/State/Zip* ______________________

**Please complete if you are not at Los Alamos.*

Please answer the following questions. Your comments help us improve the quality and usefulness of this documentation.

Are you an experienced computer user? yes _____ no _____

Is the information in this document easy to find? yes _____ no _____

Are the purpose and function of each chapter clear? yes _____ no _____

Does this document provide correct information? yes _____ no _____

Does this document fully explain the subject? yes _____ no _____

Are there enough examples in this document? yes _____ no _____

Did this document meet your expectations? yes _____ no _____

Do you think you will use this document again when your needs change? yes _____ no _____

Please use the space below (and additional pages if needed) for questions not addressed above or for expansion of responses. In particular, please explain responses marked ''no.'' When possible, please give specific page references.

____

____

____

Thank you for your help.

Figure 4